AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Currently Amended) A computer-implemented method for providing access to instrumentation data from within a managed code runtime environment, wherein the data arises from a source outside the managed code runtime environment, the method comprising:

providing an application compiled into an intermediate form from a runtime-aware programming language, the application being suitable for execution by a runtime engine in a managed code runtime environment;

executing the application in [[a]]the managed code runtime environment; having a runtime engine configured to execute applications compiled into an intermediate form, wherein there is a defined contract of operation between the executing application and the runtime engine to delegate certain application tasks to the runtime engine that enable the runtime engine to service requests from the executing application at runtime, including requests for instrumentation data representing management information about other applications and devices available in an environment outside the managed code runtime environment, the environment comprising a native code environment;

receiving a request at the runtime engine from the executing application for instrumentation data available in the from a source executing in a native code environment outside said the managed code runtime environment; , the request including a path of an instrumentation data object for accessing the instrumentation data, options used to retrieve the instrumentation data object, and an identification of a parent of the instrumentation data object;

the runtime engine binding an instance of a management object class to a particular management instrumentation object, the particular management instrumentation object representing the data source executing in the native code environment outside the managed code environment, binding the instance comprising passing a path of an instrumentation data object for accessing the instrumentation data, options used to retrieve the instrumentation data object, and an identification of a parent of the instrumentation data object;

transmitting a corresponding request for said instrumentation data to ~~an instrumentation data source existing~~ the source executing in the native code environment outside ~~said~~ the managed code runtime environment;

receiving a response to said corresponding request from said instrumentation data source;

converting said response to a format that is compatible with said managed code runtime environment; and

responding to said request for instrumentation data with said converted response.

2.   (Canceled)

3.   (Previously Presented) The method of Claim 1, wherein converting said response comprises converting the instrumentation data object to a management object that is compatible with said runtime environment.

4.   (Previously Presented) The method of Claim 3, wherein said management object encapsulates properties of the instrumentation data object accessible through said instrumentation data source, including properties representing the path of the instrumentation data object for accessing the instrumentation data, the options used to retrieve the instrumentation data object, and the identification of the parent of the instrumentation data object.

5.   (Previously Presented) The method of Claim 3, wherein said response comprises an indication that an operation was unsuccessful and wherein converting said response to said format comprises generating a management exception object.

6.   (Original) The method of Claim 5, wherein said indication that an operation was unsuccessful comprises error codes.

7.   (Currently Amended) A computer-readable storage medium comprising instructions which, when executed by a computer, cause the computer to perform the method of any_one of Claims 1 and 3-6.

8.   (Currently Amended) A computer-controlled apparatus comprising a processing unit and a system memory, and wherein the apparatus further comprises a managed code runtime environment and is configured to carry out the method of any one of Claims 1 and 3-6.

9.   (Currently Amended) A computer-implemented method for accessing instrumentation data from within a runtime environment, wherein the runtime environment provides a runtime engine that executes an application compiled from source written in a runtime-aware language into an intermediate form, the method comprising:

receiving a request from an application compiled from source written in a runtime-aware language into an intermediate form for instrumentation data representing management information about other applications and devices available in an environment outside the runtime environment, the environment comprising a native code environment, the request comprising a path of an instrumentation data object for accessing said instrumentation data, options used to retrieve the instrumentation data object, and a namespace of the instrumentation data object;

in response to said request, querying for said instrumentation data using the path of said instrumentation data object for accessing said instrumentation data;

determining whether said instrumentation data was successfully returned; [[and]]

in response to determining that said instrumentation data was successfully returned, constructing a management object in the runtime environment, the management object being constructed from an API which comprises a management object class, a management object searcher class, a management event watcher class, a management path class, a management scope class, a management exception class, and a management options class; and

populating [[said]]the management object with said instrumentation data.

10.   (Previously Presented) The method of Claim 9, wherein constructing said management object in the runtime environment and populating said management object with said instrumentation data includes binding an instance of a management object class to said instrumentation data object for accessing said instrumentation data.

11.   (Previously Presented) The method of Claim 10, further comprising constructing a management scope object identifying the namespace associated with said instrumentation data object for accessing said instrumentation data.

12.   (Previously Presented) The method of Claim 10, further comprises constructing a management path object identifying the path to said instrumentation data object.

13.   (Previously Presented) The method of Claim 10, further comprising constructing a management options object specifying the options to retrieve said instrumentation data object for accessing said instrumentation data.

14.   (Original) The method of Claim 10, further comprising:
throwing a management exception object in response to determining that said instrumentation data was not successfully returned.

15.   (Previously Presented) The method of Claim 10, wherein properties of said management object may be accessed utilizing an indexer.

16. (Currently Amended) A computer-readable storage medium comprising instructions which, when executed by a computer, cause the computer to perform the method of any one of Claims 9-15.

17.   (Previously Presented) A computer-controlled apparatus comprising a processing unit and a system memory, and wherein the apparatus further includes a managed code runtime environment and is configured to carry out the method of any one of Claims 9-15.